REMARKS

Claims 7-12 remain in this application. Claim 7 was amended in this response. No new matter has been introduced as a result of the amendment. No new matter has been introduced as a result of the amendment. Favorable reconsideration is respectfully requested.

Claims 7-12 were rejected under 35 U.S.C. §103(a) as being unpatentable over Kieckhafer et al. (U.S. Patent No. 4,805,107) in view of William (U.S. Patent No. 5,418,956). Applicants traverse the rejection for the following reasons.

Specifically, the cited art, alone or in combination, does not disclose at least one variable in a preemptively multitasking-controlled processor system including the features of "delaying the intended task change via the task scheduler when the blocking information item is input; inputting a task change information item using the task scheduler; inputting, via the currently accessing task, a release information item into the access status memory at the end of the secure access; and initiating the intended task change, via the currently accessing task, when the task change information item is input" as recited in claim 7. The claims recite a method for processing multiple tasks on one processor. The method according to the claimed invention includes a task scheduler for distributing the available time slots to different tasks. The task manager has the function to check the provided access status memory whether or not a blocking information item has been input. If no blocking information item has been input, an initiated task change is permitted by the task scheduler. Accordingly, the task scheduler only manages the distribution of the available time slots of one single processor.

Regarding Kieckhafer, the document does not teach a method for secure access to at least one variable in a preemptively multitasking-controlled processor system as recited in the present claims. Instead, Kieckhafer teaches a task scheduler for the operations controller of a multiple node fault tolerant processing system capable of processing a set of application tasks. Each node in the fault tolerant processing system has an applications processor for executing a predetermined subset of the set of application tasks and an operations controller for controlling the operation of the node and scheduling the order in which the individual tasks in the predetermined set of tasks are to be executed by the applications processor through the exchange of inter-node messages with the other nodes in the system (col. 2, lines 34-45; col. 89, lines 12-

30). In other words, Kieckhafer teaches a method for distributing multiple tasks out of a set of tasks to one task processing node out of a plurality of task processing nodes, while the present claims recite a method for secure access to at least one variable that ensures that only one active task at a time has the right to get access to the at least one variable.

Kieckhafer discloses the task scheduler having a task activities list containing an entry for each active task in the multiple node processing system, with each entry containing an execution periodicity and a node allocation, along with a priority scan list containing a list of tasks in their preferred order of execution, a completion status list containing an entry for each task stored in the priority scan list, and a selection queue storing for each node the task ready for selection in their preferred order of execution (col. 2, lines 55-63; col. 6, line 39-col. 7, line 12). The task scheduler also includes a task selector means for selecting for its own node the highest priority task currently stored in the selection queue as the next task scheduled for execution by its own applications processor, a task interactive consistency handler for updating the status of each task in the task activity list, the priority scan list, the task completed list and the selection queue which are identified in inter-node messages reporting the completion of a task (col. 3, lines 6-15). Accordingly, the configuration in Kieckhafer teaches a priority based, data driven task scheduler for selecting the task to be executed by its own applications processor to more than one processing node. The operation of the task scheduler of Kieckhafer, discussed above, ensure that every single task is processed, but only once, and only by one processing node out of a plurality of processing nodes. Kieckhafer does not teach additional information about processing multiple tasks on one single processing node.

Furthermore, Kieckhafer does not disclose a "blocking information item" as recited in the present claims. In general, the blocking information item relates to a means for indicating that an access to a variable initialized by a user task should be handled as a secure access, and as a result no task change should occur during this secure access. In contrast, Kieckhafer only discloses a message that indicates whether an input register is empty or not (col. 16, lines 19-22), or alternately, whether a message is being transmitted in relation to a completion of a task (col. 90, lines 1-8). If the Examiner is broadening the interpretation of "blocking information item" as applied to the Kieckhafer document, Applicant respectfully requests that any subsequent office action clearly define which element in Kieckhafer is being used to allegedly meet this limitation.

Furthermore, William does not solve the deficiencies of Kiekhafer, discussed above. The Office Action relied on William to purportedly teach a secure access system (col. 3, lines 5-12). However, as stated previously, this passage, merely discloses to use different "privilege levels to help ensure a secure operating environment for each task." As such, each task has access to a certain limited set of computer resources depending on the privilege level it is executed in. In no way, however, does the William reference teach or suggest a method for secure access as per the claimed invention which includes, for example, the last six method steps of independent claim 7.

Additionally, Applicant submits that there is no teaching, suggestion or motivation for one having ordinary skill in the art to combine *Kikhafer* with *William* in the manner suggested in the Office Action. The Patent Office has the initial burden of proving a *prima facie* case of obviousness. *In re Rijckaert*, 28 U.S.P.Q. 2d 1955, 1956 (Fed. Cir. 1993). In making this determination, the question is not whether the differences between the prior art and the claims themselves would have been obvious, but whether the claimed invention as a whole would have been obvious. *Stratoflex, Inc. v. Aeroquip Corp.*, 218 U.S.P.Q. 871 (Fed. Cir. 1983)(emphasis added). The initial burden is on the examiner to provide some suggestion of the desirability of doing what the inventor has done. "To support the conclusion that the claimed invention is directed to obvious subject matter, either the references must expressly or impliedly suggest the claimed invention or the examiner must present a convincing line of reasoning as to why the artisan would have found the claimed invention to have been obvious in light of the teachings of the references." *Ex parte Clapp*, 227 USPQ 972, 973 (Bd. Pat. App. & Inter. 1985). When the motivation to combine the teachings of the references is not immediately apparent, it is the duty of the examiner to explain why the combination of the teachings is proper. *Ex parte Skinner*, 2 USPQ2d 1788 (Bd. Pat. App. & Inter. 1986). (see MPEP 2142).

Further, the Federal Circuit has held that it is "impermissible to use the claimed invention as an instruction manual or 'template' to piece together the teachings of the prior art so that the claimed invention is rendered obvious." *In re Fritch*, 23 U.S.P.Q.2d 1780, 1784 (Fed. Cir. 1992). "One cannot use hindsight reconstruction to pick and choose among isolated disclosures in the prior art to deprecate the claimed invention" *In re Fine*, 837 F.2d 1071 (Fed. Cir. 1988).

Moreover, the Federal Circuit has held that "obvious to try" is not the proper standard under 35 U.S.C. §103. *Ex parte Goldgaber*, 41 U.S.P.Q.2d 1172, 1177 (Fed. Cir. 1996). "An-obvious-to-try situation exists when a general disclosure may pique the scientist curiosity, such that further investigation might be done as a result of the disclosure, but the disclosure itself does not contain a sufficient teaching of how to obtain the desired result, or that the claim result would be obtained if certain directions were pursued." *In re Eli Lilly and Co.*, 14 U.S.P.Q.2d 1741, 1743 (Fed. Cir. 1990).

Under William, different privilege levels are used to help ensure a secure operating environment for each task, where each task has access to a certain limited set of computer resources depending on the privilege level it is executed in. It is lost on the Applicants how this teaching in any way relates to the task scheduler configuration disclosed in Kieckhafer or how it could possibly be implemented as a result.

In light of the above, Applicant respectfully submits that the rejections are improper, and independent claim 7 of the present application, as well as claims 8-12 which respectfully depend therefrom, are both novel and non-obvious over the art of record. Accordingly, Applicant respectfully requests that a timely Notice of Allowance be issued in this case.

It is further submitted that no fees are due in connection with this response at this time. If any fees are due in connection with this application as a whole, the office is hereby authorized to deduct said fees from Deposit Account No.: 02-1818. If such a deduction is made, please indicate the Attorney Docket Number (0112740-271) on the account statement.

Respectfully submitted,

BELL, BOYD & LLOYD LLC

BY Peter Zura

Peter Zura
Reg. No. 48,196
P.O. Box 1135
Chicago, Illinois 60690-1135
Phone: (312) 807-4208

Dated: July 19, 2005